

Workshop

Fakta

Kurs:	Systemutvecklare C/C++ Extended 2024
Klass:	SUVx24
Teknikområde:	Programspråket C, utmatning

Learning Target

Programmering i C och C++

Grundläggande begrepp inom programmering.

Skapa källkod med C syntax.

Grundläggande in- och utmatning

Utmatning av resultat från ett program.

Användning av utvecklingsmiljöer

Använda *Visual Studio Code* (förkortat VSC) för att skapa källkod, kompilera och exekvera program skapat med programspråket C.

Introduktion

När vi skapar källkoden för våra lösningar på programmeringsproblem gör vi det som en **sekvens**. Det betyder att ett datorprogram ska följa **stegvisa instruktioner** och alltid utföra dem i ordningen **uppfifrån och ned**.

Förberedelse

Skapa en ny mapp (katalog) i den projektmapp du använder för kodning (som vi skapade tillsammans under installationen), typ:

```
C:\Chas\SUVx24\Workshop2\
```

Starta VSC och välj i menyn:

File → Open folder...

Sök fram till den nyligen skapade mappen och välj den.

Workshop

Innehåll

Repetition

Vi upprepar avslutningen från föregående Workshop.

I VSC välj i menyn

File → New file...

Och skriv filnamnet:

`helloworld.c`

Observera att du bör använda bara små bokstäver för namnet och det får inte finnas något mellanslagstecken i det heller.

Nu har du en tom textfil som är associerad med programspråket C.

Skapa källkod (koda)

Skriv in (kopiera inte) följande kod i textfilen:

```
#include <stdio.h>

void main()
{
    ....printf("Hello World");
}
```

Varför får jag inte använda Copy-Paste?

När du är ny på ett programspråk är det bra att få in syntaxen genom att skriva den för hand. Hjärnan och muskelminnet samarbetar då och du får till slut en automatik i kodandet och en känsla för hur koden ska se ut när den är korrekt.

Efter ett tag när du skrivit ungefär samma typ av kod kan du gå över och använda kopieringstekniken för att snabba upp kodandet.

Om VSC inte förstår första raden med `#include <stdio.h>` och markerar den med en röd vågig linje under:

- Vänsterklicka med musen på den rödmarkerade texten och välj översta alternativet för att göra inställningar för denna projektmapp.
- Ändra **Compiler path** till gcc, typ:

`C:/msys64/ucrt64/bin/gcc.exe`

Ändra **IntelliSense** mode till

`windows-gcc-x64`

Workshop

Du kan återkalla denna inställningssida genom att skriva i sökfältet längst upp

```
>C/C++ edit: configurations
```

Och välja (UI)-varianten, inte JSON.

Kompilera och exekvera

Provkör programmet genom att trycka på pilen till höger om 'Play'-knappen i övre högra hörnet av VSC. Välj alternativet:

```
Run C/C++ File
```

Nu bör en massa text matas ut i fönstren nedanför koden. När det stannat upp kan du byta till fliken TERMINAL och se om utskriften 'Hello World' visas (under en blåaktigt text). Om den gör det har du lyckats köra ett C-program med VSC!

Uppgift 1

Att göra:

- Skapa en ny källkodsfil för ett program i C.
- Skriv koden som löser uppgiften
- Testkör att det fungerar som tänkt (Kompilera och exekvera)
- Rätta eventuella fel och provkör igen

Skapa en ny textfil i VSC med namnet

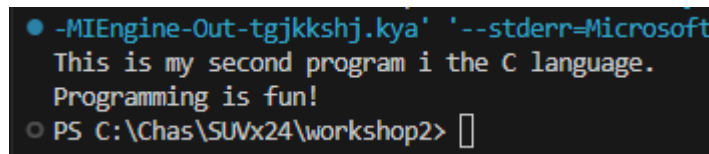
```
mysecondprogram.c
```

Skriv in följande kod:

```
#include <stdio.h>

void main()
{
    printf("This is my second program i the C language.");
    printf("\nProgramming is fun!");
}
```

Resultatet bör bli som på bilden



```
● -MIEngine-Out-tgjjkshj.kya' '--stderr=Microsoft
This is my second program i the C language.
Programming is fun!
○ PS C:\Chas\SUVx24\workshop2> □
```

Vad gör symbolerna `\n` för nytta? Kan de placeras på något annat ställe och fortfarande ge samma resultat?

Hur kan du skriva om koden med bara EN `printf`-rad?

Workshop

Uppgift 2

Att göra:

- Skapa en ny källkodsfil för ett program i C.
- Skriv koden som löser uppgiften
- Testkör att det fungerar som tänkt (Kompilera och exekvera)
- Rätta eventuella fel och provkör igen

Skapa en ny textfil i VSC med namnet

`about_me.c`

Skriv koden så att en liknande utmatning presenteras i TERMINAL-fliken. Erätt personuppgifterna med dina egna och lägg till en helt ny rad med något intressant om dig själv:

```
My name is Peter
I'm currently learning how to program in C
My pet dog is named Fido
I am 45 years old
PS C:\Chas\SUVx24\workshop2>
```

Uppgift 3a

Att göra:

- Skapa ett nytt körbart program i C.
- Testkör programmet

Använd filnamnet

`age.c`

Skriv koden:

```
#include <stdio.h>

void main()
{
    int age = 18;
    printf("My age is %d", age);
}
```

Ändra siffran till din egen ålder.

`int age`

är en variabelninitiering som ger oss ett sätt att spara datavärden i programmet och använda dem senare.

Workshop

Uppgift 3b

Att göra:

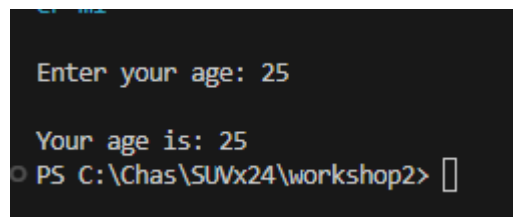
- Skapa ett nytt körbart program i C.
- Testkör programmet

Använd filnamnet

`ageinput.c`

Utgå från föregående program och skapa ett bytt program som tar emot input från den som kör programmet. Använd webbsidan [C User Input](#) för att få inspiration och exempel på hur du kan lösa detta.

En programkörning skulle kunna se ut som följande:



```
Enter your age: 25  
Your age is: 25  
PS C:\Chas\SUVx24\workshop2>
```

Bra tips

Gå till webbsidan [Int Main in C vs. Void Main in C](#) och läs på om vilken variant som är "best practise".